User Manual - English

ServerView Suite

Scripting Toolkit for Linux V1.7

User Guide

September 2015 Edition

© Copyright Fujitsu Technology Solutions GmbH 2015

All hardware and software names used are trademarks of their respective manufacturers.

All rights, including rights of translation, reproduction by printing, copying or similar methods, in part or in whole, are reserved.

Offenders will be liable for damages.

All rights, including rights created by patent grant or registration of a utility model or design, are reserved.

Delivery subject to availability. Right of technical modification reserved.

Contents

1 Preface 5

2 Scripting Toolkit Linux 7
2.1 About This Guide 7
2.2 Target Audience 7
2.3 Safety Notes 7
2.4 Notational Conventions 8

3 Overview 9
3.1 Delivery / Installation / Deinstallation 9
3.2 The Linux Deployment Environment 10
3.2.1 The Initial Ramdisk for ServerView Scripting Toolkit 10
3.2.2 Configure a NFS Server 10
3.2.3 Preparing the ServerView Scripting Toolkit repository 11

4 ServerView Scripting Toolkit Linux 13
4.1 The repository structure 13
4.1.1 Script collection 13
4.1.2 Hardware profiles 13
4.1.3 Log files 13
4.1.4 Tools 13
4.1.5 BMC Configuration 14
4.2 Deployment Process 14
4.2.1 Sample Script Structure 14

5 Setup of the Linux Deployment Environment 19
5.1 Customize the Initial Ramdisk 19
5.1.1 Unpack the Initial Ramdisk from ServerView Installation Manager 20
5.1.2 Insert STK start and configuration files 21
5.1.3 Add the Kernel Objects 22
5.1.4 Generate a customized Initial Ramdisk 22
5.2 Use the customized Initial Ramdisk to boot from CD/DVD 23
5.2.1 Using RedHat Enterprise Linux 23
5.2.2 Using SUSE Linux Enterprise Server 24
5.3 Use the customized Initial Ramdisk in PXE boot environment 25
5.3.1 Using RedHat Enterprise Linux 25

5.3.2 Using SUSE Linux Enterprise Server 26
5.4 Prepare the STK repository 27
5.4.1 Add PRIMERGY Driver Module Packages 27
5.4.2 Add Linux system libraries 28
5.4.3 Add Linux Utilities 35
5.4.4 Customize the iRMC configuration file 36
5.5 Setting up a NFS Server 36

6 Example: Configuring and Installing a PRIMERGY Server 39
6.1 Unpack ServerView_Scripting_Toolkit.tgz in <reposRoot> 39
6.2 Customize the Initial Ramdisk 39
6.2.1 Unpack Initial Ramdisk from ServerView IM 39
6.2.2 Customize the Initial Ramdisk 40
6.2.3 Add the kernel objects to the Initial Ramdisk 41
6.2.4 Generate the customized intitial ramdisk 41
6.3 Adapt the PXE Server 41
6.4 Generate the ISO image for a bootable CD 42
6.5 Prepare the STK Repository 43
6.5.1 Add PRIMERGY Kernel Modul Packages 43
6.5.2 Add Linux System Libraries 43
6.5.3 Add Linux Utilities 44

7 ServerView Utilities 47

8 Addendum 49
8.1 Supported Operating Systems 49
8.2 Supported PRIMERGY Systems 49
8.3 Released Utilities Versions 51
8.4 Restrictions 51

1 Preface

The ServerView Scripting Toolkit Linux is a collection of Linux-based utilities and sample scripts for the configuration and installation of PRIMERGY systems.

The toolkit's tools and scripts can be used to configure and install a PRIMERGY system by replicating the hardware configuration of a PRIMERGY reference system or by configuring the system's hardware directly. Both RedHat Enterprise Linux and SuSE Linux Enterprise Server installation are supported.

This guide describes the utilities provided in this toolkit and how they can be employed in a scripting environment to automate the configuration and installation process of PRIMERGY systems. It is intended as a reference for using the ServerView Scripting Toolkit for deployment of PRIMERGY servers.

The ServerView Scripting Toolkit contains:

- A collection of controller configuration utilities
- A collection of documented deployment scripts for an easy start
- Documentation
- A description on how to generate your own Linux deployment environment

The ServerView Scripting Toolkit is located

- on the PRIMERGY ServerView Installation Manager (ServerView IM) DVD in the directory <DVD Root>/SVSSoftware/Software/Deployment/PRIMERGY Scripting

or

- on Fujitsu Technology Solutions' support website
  http://download.ts.fujitsu.com/prim_supportcd/start.html
  follow *ServerView Suite – Deployment Tools*.

Things you need before getting started:

- ServerView Suite V11.15.07. or later

2 Scripting Toolkit Linux

2.1 About This Guide

This user guide is intended as a reference for using the ServerView Scripting Toolkit Linux for deployment of PRIMERGY servers.

2.2 Target Audience

This guide is targeted at experienced administrators who are familiar with the servers of the PRIMERGY family from Fujitsu Technology Solutions and have a good knowledge of Linux and the installation guides of RedHat Enterprise Linux or SUSE Linux Enterprise Server.

2.3 Safety Notes

Caution!

Improper use of the toolkit utilities can result in loss of critical data.

Because of the potential risk of data loss, only experienced individuals should use the toolkit utilities. Before using the toolkit, you must take all necessary precautions to ensure that mission-critical systems remain online if a failure occurs.

2.4 Notational Conventions

The following notational conventions are used in this manual:

`Typewriter text`	Indicates elements of the syntax that must be entered exactly as shown in the 'Syntax' section.
Italics	Indicates variables, arguments or file names
[ ]	Indicates an optional entry (an option or argument enclosed in square brackets may be specified), i.e. the entry is not mandatory.
|	Indicates an alternative option (none, one or more of several options or arguments can be specified).
...	Indicates that one or more options or arguments can be specified several times.
Bold	Used for highlighting in the text.
"Quotes"	Indicates cross-references to other chapters or manuals.
	Warning sign indicating that your health, the correct functioning of your system, or the security of your data may be at risk if you ignore the information given at this point.
	Indicates additional information, notes and hints.

3 Overview

The scripting toolkit for Linux (ServerView Scripting Toolkit) offers a collection of tools and scripts to configure the hard disks and other components before installing Linux on a PRIMERGY server. The script collection can configure

- RAID arrays
- BIOS settings
- iRMC settings

for a single or several PRIMERGY servers with same hardware configuration.

The ServerView Scripting Toolkit is intended to enable users

- to duplicate an existing reference sever configuration to several identical target servers or
- to configure any number of servers directly.

3.1 Delivery / Installation / Deinstallation

The ServerView Scripting Toolkit is delivered as a zipped tar file with the name

ServerView_Scripting_Toolkit_<version>.tgz

To install the ServerView Scripting Toolkit unpack the *.tgz* file into a folder of your choice. The following folders will be created:

doc
hwProf
scripts32
svimDVD
tools32
tools64

This folder is called the ServerView Scripting Toolkit repository and in this documentation will be referred to as *<reposRoot>.*

For de-installation just remove the ServerView Scripting Toolkit directories.

3.2 The Linux Deployment Environment

3.2.1 The Initial Ramdisk for ServerView Scripting Toolkit

A specifically prepared initial ramdisk on a PXE Server or in the installation media of the Linux distribution (bootable ISO image) is used to boot each server with the Linux operating system to access the configuration tools of ServerView Scripting Toolkit. When the boot process completes, the scripts and tools of ServerView Scripting Toolkit perform the required configuration without manual intervention.

The basis for the initial ramdisk used is provided with the ServerView Installation Manager DVD. The ServerView IM ramdisk contains the appropriate driver files for all PRIMERGY servers. You add the start script files of ServerView Scripting Toolkit and a SLES autoyast profile or a RHEL kickstart file to the ramdisk to start the system's hardware configuration and installation.

See chapter 5.1 Customize the Initial Ramdisk for details on how to customize the initial ramdisk and how to use this ramdisk with a bootable CD-ROM or via a PXE boot.

3.2.2 Configure a NFS Server

A Linux system configured as an NFS server is required for the ServerView Scripting Toolkit repository. The directory with the path <*reposRoot*> must be exported for access from NFS clients.

Then a PRIMERGY server booted using the customized initial ramdisk can mount the directory <*reposRoot*> directly after the boot process and the scripts from the ServerView Scripting Toolkit contained in the repository are executed on this PRIMERGY server.

The ServerView Scripting Toolkit scripts invoke the tools for configuration while log files within the repository provide information about the configuration process.

3.2.3 Preparing the ServerView Scripting Toolkit repository

The Linux kernel and the customized initial ramdisk starting from the bootable CD-ROM or via a PXE boot provide a working, but for toolkit purposes incomplete, Linux operating system. The following components must be added to the ServerView Scripting Toolkit repository for the configuration tools to access:

- Linux system libraries and
- Linux utilities

In addition the PRIMERGY Kernel Modul Packages must be added to the ServerView Scripting Toolkit repository. Otherwise drivers for RAID controllers are not installed and the system fails to boot after installation.

4 ServerView Scripting Toolkit Linux

4.1 The repository structure

4.1.1 Script collection

The folder *scripts32* contains a collection of sample scripts which is intended to help you start your own deployment process. The sample scripts are documented in detail in the comments of the scripts. Change or add to these sample scripts to reflect your own environment.

4.1.2 Hardware profiles

The folder *hwProf* in the toolkit repository holds all files needed for the hardware configuration during the deployment process, where hwProf */<SystemName>* contains the hardware-relevant files for the system.

4.1.3 Log files

The scripts use the *log* folder in the repository to create status log files which are intended to monitor the installation process. For each PRIMERGY system a machine-specific subfolder (using the system's serial number) will be created, which will contain the log files and progress state during the deployment process.

4.1.4 Tools

All utilities distributed with the ServerView Scripting Toolkit for Linux are located in the folder *tools32* or *tools64*.

4.1.5 BMC Configuration

To read or configure a PRIMERGY's Baseboard Management Controller (BMC) a general configuration file *stkScu.cfg* is used. All configuration values defined in the SCCI specification (see SCCI_CS.pdf in the Manuals folder), may be configured using this configuration file.

To read or configure a BMC do the following:

- Place *stkScu.cfg* in folder *hw_prof/<SysName>*
- Edit *stkScu.cfg* according to your needs.

See *scu.cfg* in folder *scripts32* for documentation.

4.2 Deployment Process

The deployment process provided with the sample script collection of the ServerView Scripting Toolkit for Linux has two modes ('SAVE' and 'DEPLOY') which are intended to either save the configuration of a server or to deploy and install a target server.

4.2.1 Sample Script Structure

- `stkProfile.sh`
  Start of deployment process.
- `setEnv.sh, newSetenv.sh`
  Deployment environment configuration, see batch file for details.
- `mss.sh`
  System detection and identification.
- `ms.sh`
  Master script, deployment process control.
- `svRaid.sh`
  Script for RAID configuration, called from `ms.sh`.
- `stkBios.sh`
  Script for BIOS settings, called from `ms.sh`.
- `svSrvCtrl.sh`
  Script for configuration of management controller BMC/RSB, called from `ms.sh`.

- `stkAddOn.sh, stkPostInstall.sh`
  Scripts to manage the installation of additional software after OS installation
- `stkFunctions.sh, stkBiosFunctions.sh, stkRaidFunctions.sh`
  These files contain functions used in the script files.
- `stkKickstart.cfg, stkYastProfile.xml`
  Unattended installation example files
- `stkScu.cfg`
  SCU (BMC/RSB) configuration example file

Notes:

All scripts are documented in the comments. Please refer to the scripts for detailed information about the scripting process.
The kickstart files delivered with the ServerView Scripting Toolkit are basic examples which cannot be expected to work in all environments. Adapt if necessary.

4.2.1.1 Shell variables

The scripting process itself uses shell variables to configure the deployment environment and control the scripting actions. The following list provides a short overview over the variables and their use. All variables are defined in *setEnv.sh*. If you are using the scripting process "as is" you need only change that file.

Note:

All scripts are documented in detail in the comments. Please refer to the scripts for detailed information about the scripting process.

4.2.1.2 Execution control variables

These variables control the way the scripting process is running.

stkMode	Execution control.
stkOutput	Screen output control
stkNewSetEnv stkNewSetEnvType	'On the fly' environment reconfiguration to be able to change environment variables without the need to create a new initial ramdisk.

4.2.1.3 File, directory, and path variables

stkDirSript	Directory which contains the scripts
stkDirTools	Directory which contains the tools
stkDirTools64	Directory which contains the tools for a 64bit OS
stkDirHardware	Directory which contains the hardware profile
stkDirTmp	Directory which contains temporary output
stkLogFileName	Log file name. Path to log file is set after system identification.

4.2.1.4 System variables

stkReposLocation	IP address of NFS server and pathname of repository on NFS server being mounted
stkReposMntDir	pathname of directory, where remote repository file system is to be mounted
stkSvrSecWait	Depending on the system and the boot medium ServerView RAID (Fujitsu Technology Solution's RAID management tool) needs some time to initialize itself after it has been started. Usually 30 seconds are sufficient. Increase the wait time if the tool does not 'see' any or all RAID controllers.

4.2.1.5 General configuration variables

stkCfgBmc	Whether to configure the baseboard management controller (iRMC).
stkFileBmc	Configuration file name for BMC configuration (iRMC
stkBiosSaveRestore	Whether to save/restore the BIOS.
stkSetBiosPassword	Whether to set the BIOS password.
stkOldBiosPassword stkNewBiosPassword	**Note:** Setting no stkNewBiosPassword clears the BIOS password; if no BIOS password exists, do not set stkOldBiosPassword.

stkSetBiosBootorder	Whether to set the BIOS boot order.
stkBiosBootorder	BIOS boot order to be set

4.2.1.6 RAID configuration variables

stkRaidSaveRestore	Whether to save or restore the system's array controllers.
stkCfgRaidDirect	Whether to configure the system's array controller directly.
stkCtrlNo	The logical number of the array controller to be configured.
stkRaidType	RAID type to configure.
stkDriveNo	Number of drives to be used for the RAID array.
stkFastInit	Wether to perform a fast initialization of the configured RAID array.
stkFileRaid	Name of file which contains the RAID configuration when stkRaidSaveRestore=YES

4.2.1.7 AddOn software configuration variables

stkCfgAddOnSw	Whether to install additional software.
stkDirAddOnSw	Folder where the additional software is located within the repository.
stkSvRaidDir	Folder where ServerView RAID Manager is located within the AddOn software folder
stkSvAgentsDir	Folder where ServerView Agents are is located within the AddOn software folder

4.2.1.8 OS unattend file location

stkYastProfile	Location of the SLES unattend file within the initial ramdisk.
stkKickStart	Location of the RHEL unattend file within the initial ramdisk.

5 Setup of the Linux Deployment Environment

After installation the ServerView Scripting Toolkit main folder contains the toolkit's subdirectories *doc, hwProf, scripts32, svimDVD, tools32* and *tools64.* In this document this folder will be called the ServerView Scripting Toolkit repository and will be referred to as

<reposRoot>.

Create a folder with pathname

<work>

Here all the work will be done to create the customized initial ramdisk and the bootable ISO image containing this ramdisk.

Notes:

It is recommended to set up the Linux Deployment Environment as user 'root'.

This document assumes the initial ramdisks to be customized are the same as on ServerView IM V11.15.07. The base for these initial ramdisks may change in later versions of ServerView IM.

Check version.txt for the initial ramdisk version and adapt the customization accordingly.

5.1 Customize the Initial Ramdisk

Create the subdirectories

<work>/*adaptIniRamDir*
In this folder the initial ramdisk delivered with the ServerView Installation Manager (ServerView IM) DVD will unpacked and customized.

<work>/*SVIMdir*
This folder will hold the appropriate initial ramdisk from the ServerView Installation Manager DVD itself.

The following steps are necessary before generating a customized initial ramdisk (see next chapter):

- insert the start script files from the ServerView Scripting Toolkit into the initial ramdisk
- insert a kickstart file for RHEL or an autoyast profile file for SLES into the initial ramdisk. This file will run the ServerView Scripting Toolkit start scripts after system boot as a preinstallation script. With this kickstart file for RHEL or autoyast profile file for SLES you can define all the parameters for a complete Linux installation.

Note:

The kickstart files delivered with the ServerView Scripting Toolkit are basic examples which cannot be expected to work in all environments. Adapt if necessary.

- insert the kernel object files
  mptctl.ko
  ipmi_si.ko
  ipmi_devintf.ko
  ipmi_msghandler.ko
  into the initial ramdisk.
  These kernel objects are needed for RAID configuration with Serverview RAID Manager or the configuration of the iRMC with the ServerView Environment and Enclosure Control Daemon Control Program respectively.

5.1.1 Unpack the Initial Ramdisk from ServerView Installation Manager

On the SVIM DVD look for

Unix/Linux/[LinuxType]/[LinuxDist]/RemoteInstallation/initrd.[dist].gz

Note:

Check version.txt for the initial ramdisk version.

Copy this file to directory

<work>/*SVIMdir*

Unpack *initrd.[dist].gz* to <work>/*adaptIniRamDir* with command

```
gunzip -c <work>/SVIMdir/initrd.[dist].gz | cpio –icvBdum
```

If `gunzip` cannot unpack the initial ramdisk (eg with the SLES12 ramdisk) use another unzip tool, e.g. `p7zip –c` or `xz –dc .`

5.1.2 Insert STK start and configuration files

Create a folder *stk* at the root of <work>*/adaptIniRamDir.*

Note:

For RHEL 7 this folder must be named ***updates/stk***.

Copy *stkProfile.sh* and *setEnv.sh* and the kickstart file *stkKickstart.cfg* or the autoyast *stkYastProfile.xml* file from the repository to the *stk* or *updates/stk* (RHEL7) directory of the unpacked initial ramdisk.

In the file

<work>*/adaptIniRamDir/stk/setEnv.sh* or
<work>*/adaptIniRamDir/updates/stk/setEnv.sh* (RHEL7)

set the shell variable *stkReposLocation* to the IP address of the NFS server and the pathname of the folder where the repository resides and set the variable *stkReposMntDir* to the mount point of the repository within the initial ramdisk.

Example:

```
stkReposLocation=<FTP>:/home/user/repository
stkReposMntDir=/localRepository
```

In the file

<reposRoot>*/scripts32/stkKickstart.cfg*

or

<reposRoot>*/scripts32/stkYastProfile.xml*

set the NFS or FTP pathname for the Linux Distribution installation tree and all other parameters for an unattended kickstart installation. With the `%pre` (RHEL) respectively the `<source>` (SLES) keyword and the following command

```
2>&1 /stk/stkProfile.sh > /stk/stkMessages.log
```

the ServerView Scripting Toolkit start files will be executed after system boot.

When installing SUSE Linux Enterprise Server, add a softlink to the autoyast profile in the root folder of the initial ramdisk. Otherwise the SLES installation program cannot find the autoyast profile.

5.1.3 Add the Kernel Objects

To be able to use ServerView RAID to configure RAID controllers, the kernel object mptctl.ko is required.
For the configuration of the iRMC the IPMI kernel objects *ipmi_si.ko, ipmi_devintf.ko* and *ipmi_msghandler.ko* are required.

Unpack the package

kernel-[version].[el-version].[arch].rpm ***(RHEL)***

kernel-default-[version].[arch].rpm ***(SLES)***

from your Linux DVD and copy the kernel objects

lib/modules/[version]/kernel/drivers/char/ipmi/ipmi_si.ko
lib/modules/[version]/kernel/drivers/char/ipmi/ipmi_devintf.ko
lib/modules/[version]/kernel/drivers/char/ipmi/ipmi_ msghandler.ko

to the *stk*-directory of the unpacked initial ramdisk

<work>*/adaptIniRamDir/stk*

5.1.4 Generate a customized Initial Ramdisk

Change to directory

<work>*/adaptIniRamDir*

and generate the customized initial ramdisk with command

```
find .|cpio –H newc –o|gzip > <work>/initrd.img
```
(RHEL)

or

```
find .|cpio –H newc –o|gzip > <work>/initrd
```
(SLES)

Note:

The name of the file containing the customized initial ramdisk is changed by this command to ***initrd.img*** or ***initrd***. This name is for use on the Linux CD/DVD boot media or PXE server.

5.2 Use the customized Initial Ramdisk to boot from CD/DVD

5.2.1 Using RedHat Enterprise Linux

In the RHEL Installation Guide you find a description how to create an own CD-ROM to boot the RHEL installation program. This CD-ROM contains the files

isolinux/initrd.img
isolinux/isolinux.cfg

Replace

isolinux/initrd.img

with the customized intital ramdisk generated in file

<work*>/initrd.img*

and adapt the file

isolinux/isolinux.cfg

like this

```
default linux
label linux
  kernel vmlinuz
  append initrd=initrd.img ks=file:/stk/stkKickstart.cfg
```

5.2.2 Using SUSE Linux Enterprise Server

A bootable CD-ROM for a SLES installation contains the files

boot/[arch]/loader/initrd
boot/[arch]/loader/isolinux.cfg

Replace

boot/[arch]/loader/initrd

by your customized intital ramdisk generated in file

<work*>/initrd*

and adapt the file

boot/[arch]/loader/isolinux.cfg

like this for RHEL 6

```
default linux
label linux
  kernel linux
  append initrd=initrd autoyast=file:/stk/stkYastProfile.xml
splash=silent
```

or like this for RHEL 7

```
default linux
label linux
  kernel linux
  append initrd=initrd autoyast=file:/stk/stkYastProfile.xml
splash=silent install=<PathToInstallationSource>
```

5.3 Use the customized Initial Ramdisk in PXE boot environment

5.3.1 Using RedHat Enterprise Linux

On the PXE server you find the files

initrd.img
vmlinuz
pxelinux.cfg/default

Replace

initrd.img

with your customized intital ramdisk generated in file

<work*>/initrd.img*

Replace

vmlinuz

with

Unix/Linux/[LinuxType]/[LinuxDist]/RemoteInstallation/vmlinuz.[dist]

from the SVIM DVD.

Adapt the file

pxelinux.cfg/default

like this

```
default linux
label linux
  kernel vmlinuz
  append initrd=initrd.img ks=file:/stk/stkKickstart.cfg
```

5.3.2 Using SUSE Linux Enterprise Server

The SLES Deployment Guide contains a description how to use PXE boot for a remote installation. The following files exist on the PXE server:

initrd
pxelinux.cfg/default

Replace

initrd

with your customized intital ramdisk generated in file

<work*>/initrd*

Replace

vmlinuz

with

Unix/Linux/[LinuxType]/[LinuxDist]/RemoteInstallation/vmlinuz.[dist]

from the SVIM DVD.

Adapt the file

pxelinux.cfg/default

like this for RHEL 6

```
default linux
label linux
  kernel linux
  append initrd=initrd autoyast=file:/stk/stkYastProfile.xml
splash=silent
```

or like this for RHEL 7

```
default linux
label linux
  kernel linux
  append initrd=initrd autoyast=file:/stk/stkYastProfile.xml
splash=silent install=<PathToInstallationSource>
```

5.4 Prepare the STK repository

Note:

Check version.txt for the initial ramdisk version and adapt the preparations accordingly.

5.4.1 Add PRIMERGY Driver Module Packages

The table following shows for each supported Linux OS the archive(s) that must be copied to their target directory in the ServerView Scripting Toolkit repository from their path on the SVIM DVD: *<DVDroot>/Unix/Linux/<LinuxType><Distribution>/RemoteInstallation*

OS	name of archive	target directory in repository
SLES11 SP4 x86 SLES11 SP4 x64	kmp_3.0.101-63-default.tgz kmp_3.0.101-63-default.tgz	svimDVD/sles11sp4 svimDVD/sles11sp4/x64
SLES12 SP0 x64	kmp_3.12.28-4-default.tgz	svimDVD/sles12sp0/x64
RHEL 6.6 x86 RHEL 6.6 x64	kmp_2.6.32-504.el6.i686.tgz kmp_2.6.32-504.el6.x86_64.tgz	svimDVD/rhel66
RHEL 6.7 x86 RHEL 6.7 x64	kmp_2.6.32-573.el6.i686.tgz kmp_2.6.32-573.el6.x86_64.tgz	svimDVD/rhel67
RHEL 7.1	kmp_3.10.0-229.el7.x86_64.tgz	svimDVD/rhel71

Note:

The KMP-file(s) contain the drivers for the OS to be installed. Make sure to keep them as much up-to-date as possible.

5.4.2 Add Linux system libraries

The SLES or RHEL installation environment does not include all libraries necessary for the ServerView RAID tool. They need to be provided in the ServerView Scripting Toolkit repository as shown below.

5.4.2.1 RHEL 6 Update 6

From folder *Packages* on the OS DVD install or unpack the packages

audit-libs-2.3.7-5.*el6.i686*
compat-libstdc++-33-3.2.3-69.el6.i686
glibc-2.12-1.149.*el6.i686*
libgcc-4.4.7-11.*el6.i686*
libstdc++-4.4.7-11.*el6.i686*
ncurses-libs-5.7-3.20090208.el6.i686
nss-softokn-freebl- 3.14.3-17.el6.i686
pam-1.1.1-20.el6.*i686.rpm*

and copy the following library files to the directory

<reposRoot>*/tools32/lib/rhel66*

in the ServerView Scripting Toolkit repository.

/lib/ld-2.12.so
/lib/libaudit.so.1.0.0
/lib/libc-2.12.so
/lib/libcrypt-2.12.so
/lib/libdl-2.12.so
/lib/libfreebl3.so
/lib/libgcc_s-4.4.7-20120601.so.1
/lib/libm-2.12.so
/lib/libnss_files-2.12.so
/lib/libpam.so.0.82.2
/lib/libpam_misc.so.0.82.0
/lib/libpthread-2.12.so
/lib/libresolv-2.12.so
/lib/librt-2.12.so
/lib/libtinfo.so.5.7
/usr/lib/libstdc++.so.5.0.7
/usr/lib/libstdc++.so.6.0.13

Add the following softlinks in directory

<reposRoot>*/tools32/lib/rhel66*

```
ln -s ld-2.12.so                    ld-linux.so.2
ln -s libaudit.so.1.0.0             libaudit.so.1
ln -s libc-2.12.so                  libc.so.6
ln -s libcrypt-2.12.so              libcrypt.so.1
ln -s libdl-2.12.so                 libdl.so.2
ln -s libgcc_s-4.4.7-20120601.so.1 libgcc_s.so.1
ln -s libm-2.12.so                  libm.so.6
ln -s libnss_files-2.12.so          libnss_files.so.2
ln -s libpam.so.0.82.2              libpam.so.0
ln -s libpam_misc.so.0.82.0         libpam_misc.so.0
ln -s libpthread-2.12.so            libpthread.so.0
ln -s libresolv-2.12.so             libresolv.so.2
ln -s librt-2.12.so                 librt.so.1
ln -s libstdc++.so.5.0.7            libstdc++.so.5
ln -s libstdc++.so.6.0.13           libstdc++.so.6
ln -s libtinfo.so.5.7               libtinfo.so.5
```

5.4.2.2 RHEL 6 Update 7

From folder *Packages* on the OS DVD install or unpack the packages

audit-libs-2.3.7-5.*el6.i686*
compat-libstdc++-33-3.2.3-69.el6.i686
glibc-2.12-1.166.*el6.i686*
libgcc-4.4.7-16.*el6.i686*
libstdc++-4.4.7-16.*el6.i686*
ncurses-libs-5.7-4.20090207.el6.i686
nss-softokn-freebl- 3.14.3-22.el6.i686
pam-1.1.1-20.el6.*i686.rpm*

and copy the following library files to the directory

<reposRoot>*/tools32/lib/rhel67*

in the ServerView Scripting Toolkit repository.

/lib/ld-2.12.so
/lib/libaudit.so.1.0.0
/lib/libc-2.12.so
/lib/libcrypt-2.12.so

/lib/libdl-2.12.so
/lib/libfreebl3.so
/lib/libgcc_s-4.4.7-20120601.so.1
/lib/libm-2.12.so
/lib/libnss_files-2.12.so
/lib/libpam.so.0.82.2
/lib/libpam_misc.so.0.82.0
/lib/libpthread-2.12.so
/lib/libresolv-2.12.so
/lib/librt-2.12.so
/lib/libtinfo.so.5.7
/usr/lib/libstdc++.so.5.0.7
/usr/lib/libstdc++.so.6.0.13

Add the following softlinks in directory

<reposRoot>*/tools32/lib/rhel67*

```
ln -s ld-2.12.so                    ld-linux.so.2
ln -s libaudit.so.1.0.0             libaudit.so.1
ln -s libc-2.12.so                  libc.so.6
ln -s libcrypt-2.12.so              libcrypt.so.1
ln -s libdl-2.12.so                 libdl.so.2
ln -s libgcc_s-4.4.7-20120601.so.1 libgcc_s.so.1
ln -s libm-2.12.so                  libm.so.6
ln -s libnss_files-2.12.so          libnss_files.so.2
ln -s libpam.so.0.82.2              libpam.so.0
ln -s libpam_misc.so.0.82.0         libpam_misc.so.0
ln -s libpthread-2.12.so            libpthread.so.0
ln -s libresolv-2.12.so             libresolv.so.2
ln -s librt-2.12.so                 librt.so.1
ln -s libstdc++.so.5.0.7            libstdc++.so.5
ln -s libstdc++.so.6.0.13           libstdc++.so.6
ln -s libtinfo.so.5.7               libtinfo.so.5
```

5.4.2.3 RHEL 7 Update 1

From folder *Packages* on the OS DVD install or unpack the packages

audit-libs-2.4.1-5.el7.x86_64
glibc-2.17-78.el7.x86_64
libgcc-4.8.3-9.el7.x86_64
libstdc++-4.8.3-9.el7.x86_64

ncurses-libs-5.9-13.20130511.el7.x86_64
nss-softokn-freebl-3.16.2.3-9.el7.x86_64
pam-1.1.8-12.el7.x86_64.rpm

and copy the following library files to the directory

<reposRoot>*/tools64/lib/rhel71*

in the ServerView Scripting Toolkit repository.

/lib64/ld-2.17.so
/lib64/libaudit.so.1.0.0
/lib64/libc-2.17.so
/lib64/libcrypt-2.17.so
/lib64/libdl-2.17.so
/lib64/libgcc_s-4.8.3-20140911.so.1
/lib64/libm-2.17.so
/lib64/libnss_files-2.17.so
/lib64/libpthread-2.17.so
/lib64/libresolv-2.17.so
/lib64/librt-2.17.so
/usr/lib64/libfreebl3.so
/usr/lib64/libtinfo.so.5.9
/usr/lib64/libpam.so.0.83.1
/usr/lib64/libpam_misc.so.0.82.0
/usr/lib64/libstdc++.so.6.0.19

Add the following softlinks in directory

<reposRoot>*/tools64/lib/rhel70*

```
ln -s ld-2.17.so                    ld-linux.so.2
ln -s libaudit.so.1.0.0             libaudit.so.1
ln -s libc-2.17.so                  libc.so.6
ln -s libcrypt-2.17.so              libcrypt.so.1
ln -s libdl-2.17.so                 libdl.so.2
ln -s libgcc_s-4.8.3-20140911.so.1 libgcc_s.so.1
ln -s libm-2.17.so                  libm.so.6
ln -s libnss_files-2.17.so          libnss_files.so.2
ln -s libpam.so.0.83.1              libpam.so.0
ln -s libpam_misc.so.0.82.0         libpam_misc.so.0
ln -s libpthread-2.17.so            libpthread.so.0
ln -s libresolv-2.17.so             libresolv.so.2
ln -s librt-2.17.so                 librt.so.1
```

```
ln -s libstdc++.so.6.0.19          libstdc++.so.6
ln -s libtinfo.so.5.9              libtinfo.so.5
```

5.4.2.4 Other RHEL versions

Using the above chapters as reference, adapt the RHEL package and library versions to the RHEL version to be used.

5.4.2.5 SLES11 SP4

Install or unpack the packages

SLES 11 SP4 x86	SLES 11 SP4 x64
audit-libs-1.8-0.34.1	*audit-libs-32bit-1.8-0.34.1*
glibc-2.11.3-17.84.1	*glibc-32bit-2.11.3-17.84.1*
libstdc++33-3.3.3-11.9	*libstdc++33-32bit-3.3.3-11.9*
libstdc++6-4.8.3+r212056-2.17	*libstdc++6-32bit-4.8.3+r212056-2.17*
libfreebl3-3.17.3-0.8.11	*libfreebl3-32bit-3-3.17.3-0.8.11*
libgcc_s1-4.8.3+r212056-2.17	*libgcc_s1-32bit-4.8.3+r212056-2.17*
libopenssl0_9_8-0.9.8j-0.70.1	*libopenssl0_9_8-0.9.8j-0.70.1*
pam-1.1.5-0.15.9	*pam-32bit-1.1.5-0.15.9*

and copy the following library files to the directory

SLES 11 SP4 x86	SLES 11 SP4 x64
<reposRoot>/tools32/lib/sles11sp4	<reposRoot>/tools64/lib/sles11sp4

in the ServerView Scripting Toolkit repository

lib/ld-2.11.3.so
lib/libaudit.so.0.0.0
lib/libc-2.11.3.so
lib/libcrypt-2.11.3.so
lib/libdl-2.11.3.so
lib/libfreebl3.so
lib/libgcc_s.so.1
lib/libm-2.11.3.so
lib/libnsl-2.11.3.so
lib/libnss_compat-2.11.3.so
lib/libnss_files-2.11.3.so
lib/libpam.so.0.83.1

lib/libpam_misc.so.0.82.0
lib/libpamc.so.0.82.1
lib/libpthread-2.11.3.so
lib/libresolv-2.11.3.so
lib/librt-2.11.3.so
usr/lib/libcrypto.so.0.9.8
usr/lib/libstdc++.so.5.0.7
usr/lib/libstdc++.so.6.0.19

Add the following softlinks in directory

SLES 11 SP4 x86	SLES 11 SP4 x64
<reposRoot>/tools32/lib/sles11sp4	<reposRoot>/tools64/lib/sles11sp4
`ln -s ld-2.11.3.so`	`ld-linux.so.2`
`ln -s libaudit.so.0.0.0`	`libaudit.so.0`
`ln -s libc-2.11.3.so`	`libc.so.6`
`ln -s libcrypt-2.11.3.so`	`libcrypt.so.1`
`ln -s libdl-2.11.3.so`	`libdl.so.2`
`ln -s libm-2.11.3.so`	`libm.so.6`
`ln -s libnsl-2.11.3.so`	`libnsl.so.1`
`ln -s libnss_compat-2.11.3.so`	`libnss_compat.so.2`
`ln -s libnss_files-2.11.3.so`	`libnss_files.so.2`
`ln -s libpamc.so.0.82.1`	`libpamc.so.0`
`ln -s libpam_misc.so.0.82.0`	`libpam_misc.so.0`
`ln -s libpam.so.0.83.1`	`libpam.so.0`
`ln -s libpthread-2.11.3.so`	`libpthread.so.0`
`ln -s libresolv-2.11.3.so`	`libresolv.so.2`
`ln -s librt-2.11.3.so`	`librt.so.1`
`ln -s libstdc++.so.5.0.7`	`libstdc++.so.5`
`ln -s libstdc++.so.6.0.19`	`libstdc++.so.6`

5.4.2.6 SLES12 SP0

Install or unpack the packages

glibc-2.19-17.72
libstdc++6-4.8.3+r212056-6.3
libstdc++6-4.8.3+r212056-6.3
libfreebl3-3.16.4-5.2
libgcc_s1-4.8.3+r212056-6.3

libopenssl1_0_0-1.0.1i-2.12
pam-1.1.8-11.57

and copy the following library files to the directory

<reposRoot>/tools64/lib/sles12sp0

in the ServerView Scripting Toolkit repository

lib64/ld-2.19.so
lib64/libc-2.19.so
lib64/libcrypt-2.19.so
lib64/libdl-2.19.so
lib64/libfreebl3.so
lib64/libgcc_s.so.1
lib64/libm-2.19.so
lib64/libnsl-2.19.so
lib64/libnss_compat-2.19.so
lib64/libnss_files-2.19.so
lib64/libpam.so.0.83.1
lib64/libpam_misc.so.0.82.0
lib64/libpamc.so.0.82.1
lib64/libpthread-2.19.so
lib64/libresolv-2.19.so
lib64/librt-2.19.so
usr/lib64/libstdc++.so.6.0.19
usr/lib64/libstdc++.so.6.0.19

Add the following softlinks in directory

<reposRoot>/tools64/lib/sles12sp0

```
ln -s ld-2.19.so              ld-linux.so.2
ln -s libc-2.19.so            libc.so.6
ln -s libcrypt-2.19.so        libcrypt.so.1
ln -s libdl-2.19.so           libdl.so.2
ln -s libm-2.19.so            libm.so.6
ln -s libnsl-2.19.so          libnsl.so.1
ln -s libnss_compat-2.19.so   libnss_compat.so.2
ln -s libnss_files-2.19.so    libnss_files.so.2
ln -s libpamc.so.0.82.1       libpamc.so.0
ln -s libpam_misc.so.0.82.0   libpam_misc.so.0
ln -s libpam.so.0.83.1        libpam.so.0
ln -s libpthread-2.19.so      libpthread.so.0
ln -s libresolv-2.19.so       libresolv.so.2
```

```
ln -s librt-2.19.so          librt.so.1
ln -s libstdc++.so.6.0.19    libstdc++.so.5
ln -s libstdc++.so.6.0.19    libstdc++.so.6
```

5.4.2.7 Other SLES versions

Using the above chapters as reference, adapt the SLES package and library versions to the SLES version to be used.

5.4.3 Add Linux Utilities

The table following shows for each supported Linux OS the files that must be extracted from the specific package and copied to their target directory in the ServerView Scripting Toolkit.

OS	Package	extract files	target directory in the ServerView Scripting Toolkit repository
SLES11 SP4 x86	coreutils-8.12-6.25.32.33.1.i586.rpm util-linux-2.19.1-6.72.1.i586.rpm zip-3.0-4.9.1.i586.rpm	id which zip	tools32/lnxUtils/sles11sp4
SLES11 SP4 x64	coreutils-8.12-6.25.32.33.1.x86_64.rpm util-linux-2.19.1-6.72.1.x86_64.rpm zip-3.0-4.9.1.x86_64.rpm	id which zip	tools64/lnxUtils/sles11sp4
SLES12 SP0 x64	coreutils-8.22-5.17.x86_64.rpm which-2.20-3.180.x86_64.rpm zip-6.00-25.70.x86_64.rpm unzip-6.00-25.70.x86_64.rpm	id which zip unzip	tools64/lnxUtils/sles12sp0
RHEL 6.6 x86	coreutils-8.4-37.el6.i686.rpm which-2.19-6.el6.i686.rpm zip-3.0-1.el6.i686.rpm unzip-6.0-1.el6.i686.rpm	id which zip unzip	tools32/lnxUtils/rhel66
RHEL 6.6 x64	coreutils-8.4-37.el6.x86_64.rpm which-2.19-6.el6.x86_64.rpm zip-3.0-1.el6.x86_64.rpm unzip-6.0-1.el6.x86_64.rpm	id which zip unzip	tools64/lnxUtils/rhel66

RHEL 6.7 x86	coreutils-8.4-37.el6.i586.rpm which-2.19-6.el6.i686.rpm zip-3.0-1.el6.i686.rpm unzip-6.0-1.el6.i686.rpm	id which zip unzip	tools32/lnxUtils/rhel67
RHEL 6.7 x64	coreutils-8.4-37.el6.x86_64.rpm which-2.19-6.el6.x86_64.rpm zip-3.0-1.el6.x86_64.rpm unzip-6.0-1.el6.x86_64.rpm	id which zip unzip	tools64/lnxUtils/rhel67
RHEL 7.1	coreutils-8.22-11.el7.x86_64.rpm which-2.20-7.el7.x86_64.rpm zip-3.0-10.el7.x86_64.rpm unzip-6.0-13.el7.x86_64.rpm	id which zip unzip	tools64/lnxUtils/rhel71

5.4.4 Customize the iRMC configuration file

To configure the integrated Remote Management Controller (iRMC) for several PRIMERGY servers (e.g. the IP address), an overall configuration file can be used. You can find an example file with the name

stkScu.cfg

in the repository folder

script32

Adapt this file to your needs and copy it to the directory

<reposRoot>*/hwProf/*<chassisType>.

See *stkScu.cfg* for documentation.

5.5 Setting up a NFS Server

On the system with the repository directory with pathname

<reposRoot>

the NFS-service must be running and the directory

<reposRoot>

must be released for access from NFS clients, so that the PRIMERGY Server started from the customized CD/DVD image or PXE server will be able to mount the directory <reposRoot> when the boot process is completed. It then can start the scripts from the ServerView Scripting Toolkit residing in the repository.

6 Example: Configuring and Installing a PRIMERGY Server

This example shows in detail how to prepare a PXE boot and how to create an ISO image to boot from CD with the customized initial ramdisk from ServerView Installation Manager for an installation of SLES12 SP0 x64 with an autoyast profile. The ServerView Scripting Toolkit start files are called from the autoyast profile as a preinstallation script.

6.1 Unpack ServerView_Scripting_Toolkit.tgz in <reposRoot>

```
export REPOSROOT=/stk/test/repository ### pathname of <reposRoot>
mkdir -p $REPOSROOT
cd $REPOSROOT
tar -xzf ../ServerView_Scripting_Toolkit_<version>.tgz
```

6.2 Customize the Initial Ramdisk

6.2.1 Unpack Initial Ramdisk from ServerView IM

```
export WORK=/stk/customizeDir
mkdir -p $WORK/SVIMdir
```

initrd.se12.gz

is located on the SVIM DVD under the pathname

Unix/Linux/SUSE/SLES_12-X86_64/RemoteInstallation/initrd.se12.gz

Copy this file to directory and unpack

$WORK/SVIMdir

```
mkdir -p $WORK/adaptIniRamDir
cd $WORK/adaptIniRamDir
xz –dc $WORK/SVIMdir/initrd.se12.gz | cpio -icvBdum
```

6.2.2 Customize the Initial Ramdisk

```
cp /stk/test/repository/scripts32/stkProfile.sh $WORK/adaptIniRamDir/stk
cp /stk/test/repository/scripts32/setEnv.sh     $WORK/adaptIniRamDir/stk
cp /stk/test/repository/scripts32/stkYastProfile.xml \
 $WORK/adaptIniRamDir/stk
```

In file

$WORK/adaptIniRamDir/stk/setEnv.sh

set variables as follows:

```
stkReposLocation=192.168.10.4:/stk/test/repository
stkReposMntDir=/stkMnt/repos
```

In file

$WORK/adaptIniRamDir/stk/stkYastProfile.xml

set the pathname for the SLES12 SP0 installation tree, for instance:

```
install: ftp://192.168.44.25/SLES12SP0
```

Add the following softlink to the autoyast profile in the root folder of the initial ramdisk. Otherwise the SLES installation program cannot find the autoyast profile.

```
cd $WORK/adaptIniRamDir
ln -s stk/stkYastProfile.xml stkYastProfile.xml
```

Note:

The kickstart files delivered with the ServerView Scripting Toolkit are basic examples which cannot be expected to work in all environments. Adapt if necessary.

6.2.3 Add the kernel objects to the Initial Ramdisk

```
mkdir /tmp/getMptctl
cd /tmp/getMptct
wget -nH --cut-dirs=3 \
ftp://192.168.44.25/SLES12SP0/suse/x86_64/kernel-default-3.12.28-
4.x86_64.rpm
rpm2cpio kernel-default-3.12.28-4.x86_64.rpm | cpio –id
cp -p \
lib/modules/*-default/kernel/drivers/message/fusion/mptctl.ko\
lib/modules/*.x86_64/kernel/drivers/char/ipmi/ipmi_si.ko \
lib/modules/*.x86_64/kernel/drivers/char/ipmi/ipmi_devintf.ko \
lib/modules/*.x86_64/kernel/drivers/char/ipmi/ipmi_msghandler.ko\
$WORK/adaptIniRamDir/stk
```

6.2.4 Generate the customized intitial ramdisk

```
cd $WORK/adaptIniRamDir
find .|cpio -H newc -o|gzip > $WORK/adaptBootIsoDir/isolinux/initrd
```

6.3 Adapt the PXE Server

Refer to the SLES12 Documentation Chapter “11.3.3 Using PXE Boot” for information on how to set up your PXE Server.

Replace

initrd

with your customized intital ramdisk generated in file

$WORK/initrd

Replace

linux

with

Unix/Linux/SuSEISLES_12SP0-X86_64/RemoteInstallation/vmlinuz.se12

from the SVIM DVD.

Overwrite the file

pxelinux.cfg/default

with this content:

```
default linux
label linux
  kernel linux
  append initrd=initrd autoyast=file:/stk/stkYastProfile.xml
splash=silent install=ftp://192.168.44.25/SLES12SP0
```

6.4 Generate the ISO image for a bootable CD

```
mkdir -p $WORK/adaptBootLoaderDir
cd $WORK/adaptBootLoaderDir
```

Get the files for the bootable CD

```
wget -r -l0 -nH --cut-dirs=3 ftp://192.168.44.25/SLES12SP0/boot/*
```

Replace the original intital ramdisk with your customized one

```
cp $WORK/initrd $WORK/adaptBootLoaderDir/boot/x86_64/loader/initrd
```

Overwrite the file

$WORK/adaptBootLoaderDir/ boot/x86_64/loader/isolinux.cfg

with this content:

```
default linux
label linux
  kernel linux
  append initrd=initrd autoyast=file:/stk/stkYastProfile.xml
splash=silent install=ftp://192.168.44.25/SLES12SP0
```

Generate the ISO file *stkSLES12SP0.iso*

```
cd $WORK/
mkisofs -o stkSLES12SP0.iso -b boot/x86_64/loader/isolinux.bin \
 -no-emul-boot \
 -boot-load-size 4 -boot-info-table -R -J -v -T \
 adaptBootLoaderDir
```

6.5 Prepare the STK Repository

6.5.1 Add PRIMERGY Kernel Modul Packages

The archives containing the PRIMERGY Kernel Module Packages for SLES12 SP0 x86_64 are located on the SVIM DVD under the pathname

Unix/Linux/SuSE/SLES_12-X86_64/RemoteInstallation

in the file

kmp_3.12.28-4-default.tgz

Copy these files to directory

$REPOSROOT/svimDVD/sles12sp0

6.5.2 Add Linux System Libraries

Make sure that the packages

glibc-2.19-17.72
libstdc++6-4.8.3+r212056-6.3
libstdc++6-4.8.3+r212056-6.3
libfreebl3-3.16.4-5.2
libgcc_s1-4.8.3+r212056-6.3
libopenssl1_0_0-1.0.1i-2.12

are installed (e.g. using '`rpm -q <pkgname>`') and copy the following library files to the directory

$REPOSROOT/tools64/lib/sles12sp0

in the ServerView Scripting Toolkit repository.

```
cd $REPOSROOT/tools64/lib/sles12sp0
cp lib64/ld-2.19.so .
cp lib64/libc-2.19.so .
cp lib64/libcrypt-2.19.so .
cp lib64/libdl-2.19.so .
cp lib64/libfreebl3.so .
cp lib64/libgcc_s.so.1 .
cp lib64/libm-2.19.so .
```

```
cp lib64/libnsl-2.19.so .
cp lib64/libnss_compat-2.19.so .
cp lib64/libnss_files-2.19.so .
cp lib64/libpam.so.0.83.1 .
cp lib64/libpam_misc.so.0.82.0 .
cp lib64/libpamc.so.0.82.1 .
cp lib64/libpthread-2.19.so .
cp lib64/libresolv-2.19.so .
cp lib64/librt-2.19.so .
usr/lib64/libstdc++.so.6.0.19
usr/lib64/libstdc++.so.6.0.19
```

Add these softlinks

```
ln -s ld-2.19.so              ld-linux.so.2
ln -s libc-2.19.so            libc.so.6
ln -s libcrypt-2.19.so        libcrypt.so.1
ln -s libdl-2.19.so           libdl.so.2
ln -s libm-2.19.so            libm.so.6
ln -s libnsl-2.19.so          libnsl.so.1
ln -s libnss_compat-2.19.so   libnss_compat.so.2
ln -s libnss_files-2.19.so    libnss_files.so.2
ln -s libpamc.so.0.82.1       libpamc.so.0
ln -s libpam_misc.so.0.82.0   libpam_misc.so.0
ln -s libpam.so.0.83.1        libpam.so.0
ln -s libpthread-2.19.so      libpthread.so.0
ln -s libresolv-2.19.so       libresolv.so.2
ln -s librt-2.19.so           librt.so.1
ln -s libstdc++.so.6.0.19     libstdc++.so.5
ln -s libstdc++.so.6.0.19     libstdc++.so.6
```

6.5.3 Add Linux Utilities

```
mkdir /tmp/lnxUtils
cd /tmp/lnxUtils
```

Get the package files containing 'id' and 'zip' and 'which'

```
wget -q -nH --cut-dirs=5
ftp://192.168.44.25/SL12SP0/suse/x86_64/coreutils-8.22-5.17.x86_64.rpm
wget -q -nH --cut-dirs=5 ftp://192.168.44.25/SL12SP0/suse/x86_64/which-
2.20-3.180.x86_64.rpm
```

```
wget -q -nH --cut-dirs=5 ftp://192.168.44.25/SL12SP0/suse/x86_64/zip-3.0-
15.18.x86_64.rpm
```

Unpack these packages

```
rpm2cpio coreutils-8.22-5.17.x86_64.rpm | cpio –id
rpm2cpio which-2.20-3.180.x86_64.rpm | cpio –id
rpm2cpio unzip-6.00-25.70.x86_64.rpm | cpio –id
rpm2cpio zip-3.0-15.18.x86_64.rpm | cpio –id
```

Copy the packages to the repository library

```
cd usr/bin
cp id which zip $REPOSROOT/tools64/lnxUtils/sles12sp0
```

7 ServerView Utilities

For detailed information on the use of the utilities please refer to the Tools Documentation.

Caution!

Utilities marked (*) are unsupported.

Unsupported utilities are part of this release for the benefit of supporting individual customer setups. Each utility is provided as is. These utilities are not guaranteed to work in all possible configurations and must be thoroughly tested before being used in mission critical environments.

These utilities are distributed with the ServerView Scripting Toolkit:

- BIOS configuration
    - DeskView DeskFlash
    - DeskView BiosSet
    - NVRAM_Read_Write_CX400
- BMC / iRMC configuration
    - Environment and Enclosure Control Daemon (eecd)
    - Environment and Enclosure Control Daemon Control Program (eecdcp)
- RAID configuration
    - ServerView RAID (amCLI)

8 Addendum

8.1 Supported Operating Systems

- RHEL 6.6 x86/x64
- RHEL 6.7 x86/x64
- RHEL 7.1 x64
- SLES11 SP4 x86/x64
- SLES12 SP0 x64

8.2 Supported PRIMERGY Systems

Blade Systems

- BX620S6
- BX920S1, BX920S2, BX920S3, BX920S4
- BX922S2
- BX924S2, BX924S3, BX924S4
- BX960S1
- BX2560M1, BX2560M2
- BX2580M1, BX2580M2

Cloud Systems

- CX122S1
- CX250S1, CX250S2
- CX270S1, CX270S2
- CX272S1
- CX2550M1, CX2550M2
- CX2570M1, CX2570M2

Micro Systems

- MX130 S1, MX130 S2

Rack Systems

- RX100S5, RX100S6, RX100S7, RX100S7p, RX100S8
- RX200S7, RX200S8
- RX300S7, RX300S8
- RX350S7, RX350S8
- RX500S7
- RX600S6
- RX900S1, RX900S2
- RX1330M1, RX1330M2
- RX2510M1, RX2510M2
- RX2520M1, RX2520M2
- RX2530M1, RX2530M2
- RX2540M1, RX2540M2
- RX2560M1
- RX4770M1, RX4770M2, RX4770M3

Storage Systems

- SX131S1, SX150 S8, SX350 S8

Tower Systems

- TX100S1, TX100S2, TX100S3, TX100S3p
- TX120F, TX120S1, TX120S2, TX120S3, TX120S3p
- TX140S1, TX140S1p, TX140S2
- TX150S7, TX150S8
- TX200S7
- TX300S7, TX300S8
- TX1310M1, TX1310M2
- TX1320M1, TX1320M2
- TX1330M1
- TX2540M1
- TX2560M1

8.3 Released Utilities Versions

ServerView RAID (amCLI)	V6.2-8
ServerView Agents Linux (eecd, eecdcp)	V7.10-16
DeskView BIOS Management Linux (deskflash, biosset)	V1.62-0036
NVRAM_Read_Write_CX400	1.0.5

8.4 Restrictions

The following restrictions apply to V1.7 of the ServerView Scripting Toolkit:

- BIOS cannot be saved/restored or configured for the following systems :
  TX120 S2,
  RX500 S7, RX600 S5, RX900 S1, RX900 S2
  BX620 S5, BX620 S6
  BX920 S1, BX920 S2, BX922 S1, BX922 S2, BX924 S2, BX960 S1
- BIOS password cannot be set on RX100 S7
- On RHEL 7 BIOS cannot be configured for any CX server.